

Package: hrmn (via r-universe)

October 2, 2026

Title Harmonize Datasets

Version 0.0.0.9005

Description A common early step during data analysis is ``data harmonization" -- converting disparate datasets into a unified, consistent format, with consistent column names, classes, and values. The goal of 'hrmn' is to make this process as easy as it can be.

License MIT + file LICENSE

URL <https://hrmn.wrangle.zone/>, <https://github.com/wranglezone/hrmn>

BugReports <https://github.com/wranglezone/hrmn/issues>

Depends R (>= 4.1)

Imports fastmatch, rlang, stbl (>= 0.2.0.9002), tibble

Suggests testthat (>= 3.0.0)

Remotes stbl=wranglezone/stbl

Config/testthat/edition 3

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Repository <https://wranglezone.r-universe.dev>

Date/Publication 2025-10-17 12:22:35 UTC

RemoteUrl <https://github.com/wranglezone/hrmn>

RemoteRef HEAD

RemoteSha 040ffbb661d0c8f4637120b66948bddc5a10c12d

Contents

harmonize_df	2
harmonize_fct	3
specify_df	4
specify_fct	5

Index**6**

harmonize_df	<i>Harmonize a data frame</i>
--------------	-------------------------------

Description

Harmonize a data frame

Usage

```
harmonize_df(  
  .data,  
  ...,  
  .spec = NULL,  
  .unspecified_columns = c("error", "drop", "keep")  
)
```

Arguments

.data	(data.frame) A data frame to harmonize.
...	These dots are for future extensions and must be empty.
.spec	(hrmn_spec_df) A data frame harmonization specification.
.unspecified_columns	("error", "drop", or "keep") How to handle columns in .data that are not present in .spec.

Value

The input .data harmonized to a `tibble::tibble()`.

See Also

Other harmonization functions: `harmonize_fct()`

Examples

```
df <- data.frame(  
  size = c("Small", "Medium", "S", "M", "Large", "Lrg", "Sm"),  
  id = 1:7  
)  
  
# This spec will coerce values to NA if they are not "Small", "Medium",  
# or "Large".  
spec <- specify_df(  
  size = specify_fct(levels = c("Small", "Medium", "Large"))  
)  
  
# We can provide harmonization rules to the data before the spec is applied.
```

```
# Here, we harmonize the input factor to convert "S", "M", "Sm", and "Lrg" to
# valid values.
harmonize_df(
  df,
  size = harmonize_fct(
    size,
    .lookup = c("S" = "Small", "M" = "Medium", "Sm" = "Small", "Lrg" = "Large")
  ),
  .spec = spec,
  .unspecified_columns = "keep"
)
```

harmonize_fct	<i>Harmonize a factor</i>
---------------	---------------------------

Description

Harmonize a factor

Usage

```
harmonize_fct(.data, ..., .spec = NULL, .lookup = NULL)
```

Arguments

.data	(character or coercible to character) A vector to harmonize to the specified factor.
...	These dots are for future extensions and must be empty.
.spec	(hrmn_spec_fct) A harmonization specification from specify_fct() .
.lookup	(named character) A vector of replacement values. The names are the values in .data and the values are the target values.

Value

A harmonized [factor\(\)](#).

See Also

Other harmonization functions: [harmonize_df\(\)](#)

Examples

```
# Without a spec, harmonize_fct() acts like [base::factor()].
harmonize_fct(c("a", "b", "c"))

# Basic harmonization, dropping levels not in the spec
spec <- specify_fct(levels = c("a", "b"))
harmonize_fct(c("a", "b", "c"), .spec = spec)
```

```
# Using a lookup table to recode values
spec2 <- specify_fct(levels = c("fruit", "citrus"))
lookup <- c(apple = "fruit", banana = "fruit", orange = "citrus")
harmonize_fct(
  c("apple", "banana", "orange"),
  .spec = spec2,
  .lookup = lookup
)
```

specify_df	<i>Data frame specification</i>
------------	---------------------------------

Description

Create an object that specifies the desired format for a data frame. This specification object does not contain any data itself, only the rules for harmonization.

Usage

```
specify_df(...)
```

Arguments

... (hrmn_spec) Column specifications, given as named arguments.

Value

A `hrmn_spec_df` object that acts as a specification.

See Also

Other specification functions: [specify_fct\(\)](#)

Examples

```
specify_df(
  response = specify_fct(levels = c("Yes", "No", "Maybe")),
  outcome = specify_fct(levels = c("Positive", "Negative"))
)
```

specify_fct	<i>Factor specification</i>
-------------	-----------------------------

Description

Create an object that specifies the desired levels for a factor variable. This specification object does not contain any data itself, only the rules for harmonization.

Usage

```
specify_fct(levels = character())
```

Arguments

levels (character) The allowed values of the factor.

Value

A `hrmn_spec_fct` object that acts as a specification.

See Also

Other specification functions: [specify_df\(\)](#)

Examples

```
specify_fct(levels = c("a", "b", "c"))
```

Index

* **harmonization functions**

harmonize_df, [2](#)

harmonize_fct, [3](#)

* **specification functions**

specify_df, [4](#)

specify_fct, [5](#)

factor(), [3](#)

harmonize_df, [2](#), [3](#)

harmonize_fct, [2](#), [3](#)

specify_df, [4](#), [5](#)

specify_fct, [4](#), [5](#)

specify_fct(), [3](#)

tibble::tibble(), [2](#)